UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/749,851	12/30/2003	Ingo Zenz	6570P017	9437

45062 7590 10/15/2008
SAP/BSTZ
BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP
1279 OAKMEAD PARKWAY
SUNNYVALE, CA 94085-4040

EXAMINER
BROMELL, ALEXANDRIA Y

ART UNIT	PAPER NUMBER
2167	

MAIL DATE	DELIVERY MODE
10/15/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/749,851	**Applicant(s)** ZENZ, INGO	
	Examiner ALEXANDRIA Y. BROMELL	**Art Unit** 2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 May 2008*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-22* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 December 2003* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date *1/25/08, 5/31/08, 9/19/08*.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

Response to Arguments

Applicant's arguments, see Remarks, filed December 18, 2007, with respect to the rejection(s) of claims 1 – 22 have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of David Heath et al. (U.S. patent 5,553,239) and Vij Rajarajan et al. (U.S. Patent Publication 20020161750).

Information Disclosure Statement

The information disclosure statements (IDS) submitted on 1/25/08, 5/30/08, and 9/18/08 are in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statements are being considered by the examiner.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1 - 10, and 18 – 22 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 1 - 10, and 18 – 22 are rejected under 35 USC 101 for being "software per se".

The claimed invention as in claims 1 - 10, and 18 – 22 is addressed to a system made up of server nodes organize as instances, therefore, the claims are deemed to read as pure software systems, with no clear limitations that read on some sort of hardware.

In view of Applicant's disclosure, specification paragraphs [0029], the present invention may be embodied in hardware and/or software. Accordingly, the claim may become nothing more than a set of software instructions which are "software per se".

"Software per se" is non-statutory under 35 USC 101 because it is merely a set instruction without any defined tangible output or tangible result being produced. The requirement for tangible result under 35 USC 101 is defined in *State Street Bank & Trust Co. v. Signature Financial Group Inc.*, 149 F.3d 1368, 47USPQ2d 1596 (Fed. Cir. 1998).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1 – 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over David Heath et al. (U.S. patent 5,553,239), hereinafter, "Heath," in view of Vij Rajarajan et al. (U.S. Patent Publication 20020161750), hereinafter, "Rajarajan."

With respect to claim 1, Heath teaches a plurality of server nodes communicatively coupled on a network to serve applications over the network to a plurality of clients the plurality of server nodes organized as instances, each instance including at least one server process (i.e. minicomputer client nodes are represented as 52a and b, which are connected to a plurality of process server nodes, column 8, lines 9 – 20, also see figures 3 and 4), a data object to store a hierarchical representation of configuration data associated with management of the server nodes, the data object accessible by all of the servers and the hierarchy having a root, a first subset of nodes branching from the root containing configuration data associated with all servers in nodes branching from the root, a second subset of nodes branching from the root containing configuration data specific to all servers of a first instance and a third subset of nodes branching from the node of the second subset, the third subset of nodes containing configuration data specific to a first server of the first instance (i.e. entry manager 54m controls hierarchical representation of data with respect to server nodes, and represents the root, with various branching nodes, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4), a configuration manager on at least one of the server nodes to determine whether the configuration data stored on the at least one server node is out-of-date based on the location of an updated configuration parameter within the hierarchy, wherein updated configuration parameters within the first subset of nodes renders all servers out-of-date, updated configuration parameters within the second subset of nodes renders all servers of the first instance out-of-date and updated configuration parameters within the third subset of nodes renders only the first server out-of- date (i.e. entry manager 54m controls ensures reliable, continuous operation to the nodes in the hierarchy, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4).

Heath does not explicitly teach property sheets. However, Rajarajan teaches a property sheet logically positioned at one of the nodes of the data object, the property sheet data structure including a plurality of property names, wherein each respective property name included in the property sheet data structure is associated with a default configuration value and, optionally, a custom configuration value to pair the default configuration value to the custom configuration value for a configuration parameter represented in the data object, wherein the property sheet data structure preserves both the value of the default parameter and the custom configuration value (i.e. property sheets correspond to an instance of a data object, [0011], the property sheets contain names, [0081], also see figure 5).

Heath and Rajarajan are analogous art because they are from the same field of endeavor of managing information in a network. At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]). Therefore, it would have been obvious to combine Rajarajan with Heath to obtain the invention as specified in the instant claims.

With respect to claim 2, Heath teaches wherein the data object is stored within a central database accessible by each of the server nodes and a first node of the data object contains global configuration data associated with the plurality of server nodes and a second node of the data object contains configuration data specific to a one of the plurality of server nodes (i.e. process server nodes access data that is stored in a central master database 61, column 6, lines 23 – 31, see figure 3).

With respect to claim 3, Heath does not explicitly teach property sheets. However, Rajarajan teaches once the default configuration value has been modified, the default configuration value is stored independently with respect to the custom configuration values in the property sheet data structure (i.e. when configuration data has been modified, it is updated in the database as well, [0037]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 4, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the default configuration values associated with the property sheet data structure are modifiable using an interface other than a user interface (i.e. changes to the management module may effectively modify data sheet information, [0041]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 5, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the property sheet data structure is associated with a particular component or a set of components contained within a clustered system (i.e. property sheet structure is associated with components like pages, [0184]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 6, Heath does not explicitly teach property sheets. However, Rajarajan teaches a first dialog box to display contents of the property sheet data structure, the first dialog box including a plurality of entry rows, each respective entry row of the first dialog box including a first column to display names of corresponding properties, a second column to display current configuration values associated with corresponding properties and a third column to indicate if a configuration value displayed in the second column is a default configuration value or a custom configuration value (i.e. property sheet has corresponding dialog boxes with rows and columns of objects, see figure 14), and a second dialog box including a data entry field to enable a user to modify a selected default or custom value (i.e. method for modifying property sheet value, [0187], also see figure 23).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 7, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein a custom configuration value associated with a property is modifiable by selecting the second dialog box of the corresponding property and entering a new value in the data entry field of the second dialog box (i.e. method for modifying property sheet value, [0187], also see figure 23).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 8, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the second dialog box of the corresponding property is selected by clicking a custom check box inside the third column of a corresponding entry row (i.e. method for modifying property sheet value, [0187], where you may select a different user from a list given their row and column, also see figure 23).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 9, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the second dialog box further includes a name field to display a name of a corresponding property and a default field to display a default configuration value associated with the corresponding property (i.e. property sheets correspond to an instance of a data object, [0011], the property sheets contain names, [0081], also see figure 5).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 10, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the second dialog box further includes a data type field to display the data type associated with corresponding property (i.e. console allows user to choose object type, [0134], also see figure 12).

With respect to claim 11, Heath teaches storing within a data object, a hierarchical representation of configuration data associated with management of a plurality of server nodes organized as instances, each instance including at least one server process, the data object accessible by all of the servers and the hierarchy having a root, a first subset of nodes branching from the root containing configuration data associated with all servers in nodes branching from the root, a second subset of nodes branching from the root containing configuration data specific to all servers of a first instance and a third subset of nodes branching from the node of the second subset, the third subset of nodes containing configuration data specific to a first server of the first instance (i.e. entry manager 54m controls hierarchical representation of data with respect to server nodes, and represents the root, with various branching nodes, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4).

Heath does not explicitly disclose property sheets. However, Rajarajan teaches providing one or more property sheets at one or more of the nodes, each of the property sheets including a plurality of configuration parameters associated with the server nodes, each parameter associated with a name, a default configuration value and optionally a custom configuration value (i.e. property sheets correspond to an instance of a data object, [0011], the property sheets contain names, [0081], also see figure 5), pairing the default configuration value with an updated custom configuration value for a configuration parameter represented in the data object in response to the user specifying a custom parameter value wherein both the default configuration value and the custom configuration value are preserved by the property data sheet (i.e. when the system receives more than one schema document, the property sheet is modified to merge the property pages, [0011]), determining whether to invalidate the configuration data stored on another of the server nodes based on the location of the updated configuration parameters within the hierarchy, wherein updated configuration parameters within the first subset of nodes invalidates configuration data of all servers, updated configuration parameters within the second subset of nodes invalidates configuration data of all servers of the first instance and updated configuration parameters within the third subset of nodes invalidates configuration data of only the first server (i.e. protocols are established to determine if features of the resources or the configuration should be modified, [0037]).

Heath and Rajarajan are analogous art because they are from the same field of endeavor of managing information in a network. At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]). Therefore, it would have been obvious to combine Rajarajan with Heath to obtain the invention as specified in the instant claims.

With respect to claim 12, Heath teaches storing the data object, configuration data, binaries and property sheets within a central database, the central database accessible by the server nodes and a first node of the data object containing global configuration data associated with the plurality of server nodes and a second node of the data object containing configuration data specific to a one of the plurality of server nodes (i.e. process server nodes access data that is stored in a central master database 61, column 6, lines 23 – 31, see figure 3).

With respect to claim 13, Heath does not explicitly disclose property sheets. However, Rajarajan teaches opening the property sheet in a property sheet graphical user interface, the graphical user interface comprising a first column for storing parameter names, a second column for storing a current parameter value and a third column for storing an indication as to whether the current parameter value is a custom value or a default value; selecting the indication in the third column (i.e. property sheet has corresponding dialog boxes with rows and columns of objects, see figure 14, use of a graphical user interface, [0003 – 0004]), responsively generating a data entry window having a custom field for entering a custom value, receiving user entry of a custom value in the custom field (i.e. data in the fields can be entered or received, [0079]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 14, Heath does not explicitly disclose that the system used Java. However, Rajarajan teaches wherein the server nodes are Java server nodes supporting the Java 2 Enterprise Edition ("J2EE") standard and wherein the property sheet parameters comprise J2EE parameters (i.e. Java applets and scripts are used, [0127 – 0128]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 15, Heath teaches storing, a hierarchical representation of configuration data associated with management of the server nodes, the data object accessible by all of the servers and the hierarchy having a root, a first subset of nodes branching from the root containing configuration data associated with all servers in nodes branching from the root, a second subset of nodes branching from the root containing configuration data specific to all servers of a first instance and a third subset of nodes branching from the node of the second subset, the third subset of nodes containing configuration data specific to a first server of the first instance (i.e. entry manager 54m controls hierarchical representation of data with respect to server nodes, and represents the root, with various branching nodes, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4), determining if the configuration data stored on the other server nodes is out-of-date based on the location of the updated configuration parameters within the hierarchy, wherein updated configuration parameters within the first subset of nodes renders all servers out-of-date, updated configuration parameters within the second subset of nodes renders all servers of the first instance out-of-date and updated configuration parameters within the third subset of nodes renders only the first server out-of-date (i.e. entry manager 54m controls ensures reliable, continuous operation to the nodes in the hierarchy, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4).

Heath does not explicitly disclose property sheets. However, Rajarajan teaches providing one or more property sheets at one or more of the nodes, each of the property sheets including a plurality of configuration parameters associated with the server nodes, each parameter associated with a name, a default configuration value and optionally a custom configuration value (i.e. property sheets correspond to an instance of a data object, [0011], the property sheets contain names, [0081], also see figure 5), pairing the default configuration value with an updated custom configuration value for a configuration parameter represented in the data object in response to the user specifying a custom parameter value wherein both the default configuration value and the updated custom configuration value are preserved by the property data sheet (i.e. when the system receives more than one schema document, the property sheet is modified to merge the property pages, [0011]), communicating an indication of the configuration parameter update to one or more other server nodes (i.e. updates and edits are communicated to the communication manager, [0066]), downloading the updated configuration data from the central database to the other server nodes if the configuration data stored on the other server nodes is out-of-date (i.e. protocols are established to determine if features of the resources or the configuration should be modified or updated, [0037]).

Heath and Rajarajan are analogous art because they are from the same field of endeavor of managing information in a network. At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]). Therefore, it would have been obvious to combine Rajarajan with Heath to obtain the invention as specified in the instant claims.

With respect to claim 16, Heath does not explicitly teach locks. However, Rajarajan teaches acquiring a lock on the configuration parameters stored within the property sheet prior to updating the configuration parameters at the first server node (i.e. system creates a proxy which enables the system to reach information regardless of its location, [0172]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]). With respect to claim 17, Heath does not explicitly teach locks. However, Rajarajan teaches releasing the lock on the configuration parameters after the configuration data has been updated at the central database and/or at each of the other server nodes.

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 18, Heath teaches server node means communicatively coupled on a network, the server node means to serve applications over the network to a plurality of clients and the server node means organized as instances, each instance including at least one server node means (i.e. minicomputer client nodes are represented as 52a and b, which are connected to a plurality of process server nodes, column 8, lines 9 – 20, also see figures 3 and 4), hierarchical data object means to store a hierarchical representation of configuration data associated with the server node means, the hierarchical data object means accessible by the plurality of server node means and including a first subset of nodes containing configuration data associated with all server node means and a second subset of nodes containing configuration data specific to all server node means within a first instance (i.e. entry manager 54m controls hierarchical representation of data with respect to server nodes, and represents the root, with various branching nodes, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4), a configuration managing means on the server node means to determine whether the configuration data stored on the server node means is out-of-date based on a location of an updated configuration parameter within the hierarchical representation, wherein updated configuration parameters within the first subset of nodes renders all server node means out-of- date and updated configuration parameters within the second subset of nodes renders all server node means of the first instance out-of-date (i.e. entry manager 54m controls ensures reliable, continuous operation to the nodes in the hierarchy, column 3, lines 65 - 67, column 4, lines 1 – 10, also see figures 3 and 4).

Heath does not explicitly teach property sheets. However, Rajarajan teaches property sheet means logically positioned at one of the nodes of the data object, the property sheet means including a plurality of property names, wherein each respective property name included in the property sheet means is associated with a default configuration value and, optionally, a custom configuration value to pair the default configuration value with the custom configuration value for a configuration parameter represented in the hierarchical data object means, wherein the property sheet data structure preserves both the value of the default parameter and the custom configuration value (i.e. property sheets correspond to an instance of a data object, [0011], the property sheets contain names, [0081], also see figure 5).

Heath and Rajarajan are analogous art because they are from the same field of endeavor of managing information in a network. At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]). Therefore, it would have been obvious to combine Rajarajan with Heath to obtain the invention as specified in the instant claims.

With respect to claim 19, Heath teaches wherein the hierarchical data object means is stored within a central database accessible by each of the server nodes and a first node of the hierarchical data object means contains global configuration data associated with the plurality of server nodes and a second node of the hierarchical data object means contains configuration data specific to a one of the plurality of server nodes (i.e. process server nodes access data that is stored in a central master database 61, column 6, lines 23 – 31, see figure 3).

With respect to claim 20, Heath does not explicitly teach property sheets. However, Rajarajan teaches once the default configuration value has been modified, the default configuration value is stored independently with respect to the custom parameters in the property sheet means (i.e. when configuration data has been modified, it is updated in the database as well, [0037]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 21, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the custom configuration values associated with the property sheet means are not user-modifiable via the user interface (i.e. changes to the management module may effectively modify data sheet information, [0041]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

With respect to claim 22, Heath does not explicitly teach property sheets. However, Rajarajan teaches wherein the property sheet means is associated with a particular component or a set of components contained within the server node means (i.e. property sheet structure is associated with components like pages, [0184]).

At the time of the invention, it would have been obvious to one of ordinary skill in the art to modify the teachings of Heath with the system and method of Rajarajan in order to manage computer resources to provide easier, higher-level operations in a management framework (Rajarajan, [0010]). The motivation for doing so would have been to use property sheets to represent objects in a distributed network environment (Rajarajan, [0011]).

Conclusion/Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ALEXANDRIA Y. BROMELL whose telephone number is (571)270-3034. The examiner can normally be reached on M-F 8-4.

If attempts to reach the examiner by telephone are unsuccessful, the examiner’s supervisor, John R. Cottingham can be reached on 571-272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Alexandria Y Bromell/
Examiner, Art Unit 2167
October 8, 2008

/Shahid Al Alam/
Primary Examiner, Art Unit 2162